

An integration of a low cost adaptive remeshing strategy in the solution of structural shape optimization problems using evolutionary methods

Gabriel Buggeda ^{a,*}, Juan José Ródenas ^b, Eugenio Oñate ^c

^a *Escola Universitària d'Enginyeria Tècnica Industrial de Barcelona (EUETIB-UPC), C/Comte d'Urgell, 187, 08036 Barcelona, Spain*

^b *Departamento de Ingeniería Mecánica y de Materiales, Centro de Investigación de Tecnología de Vehículos, Universidad Politécnica de Valencia (UPV), Camino de Vera s/n, 46022 Valencia, Spain*

^c *International Center for Numerical Methods in Engineering (CIMNE-UPC), C/Gran Capitán s/n, Campus Nord UPC, Módulo C1, 08034 Barcelona, Spain*

Received 5 March 2007; accepted 30 April 2007

Available online 21 June 2007

Abstract

Evolutionary methods are a powerful and robust tool for the solution of structural shape optimization problems. Nevertheless, the use of these methods requires the structural analysis of an important number of different designs, this making the computational cost of the analysis of each design a critical issue. For this reason, each design must be analyzed at a minimum computational cost but ensuring a minimum quality of the results.

It is well known that the cheapest mesh for producing a solution with a fixed quality at minimum cost is an adapted one. Nevertheless, traditional adapted meshes are obtained from adaptive remeshing strategies, where each design has to be analyzed more than once, thereby also causing a high computational cost.

This work presents a new strategy that allows generating an adapted mesh for each design without the necessity of performing a full adaptive remeshing procedure for each of them. It is based on the use of sensitivity analysis of all magnitudes related with adaptive remeshing (location of nodes, error estimation, etc.) with respect to the design variables. This sensitivity analysis is performed only once using a geometry of reference and it is used to project the results of the corresponding analysis to all other designs to be analyzed. The projected information allows generating an appropriate adapted mesh for each new design in one shot, greatly reducing the computational cost compared with standard strategies.

© 2007 Elsevier Ltd. All rights reserved.

Keywords: Structural shape optimization; Adaptive remeshing; Sensitivity analysis; Evolutionary algorithms; Mesh optimality criteria; Differential evolution

1. Introduction

From a mathematical point of view, the treatment of an optimization problem can be viewed as the minimization of a function $f(\mathbf{x})$ depending on a set of variables \mathbf{x} and subjected to some constraints. The general form of such a problem is:

$$\begin{aligned} \text{minimize: } & f(\mathbf{x}); \mathbf{x} = \{x_i\}; \quad i = 1, \dots, n \\ \text{with: } & \mathbf{g}(\mathbf{x}) = \{g_j(\mathbf{x})\}; \quad j = 1, \dots, m \\ \text{verifying: } & g_j(\mathbf{x}) \leq 0; \quad j = 1, \dots, m \\ & a_i \leq x_i \leq b_i; \quad i = 1, \dots, n \end{aligned}$$

where f is the objective function, x_i are the design variables and g_j are inequality constraints which, for structural problems, are normally expressed in terms of stresses or displacements. The values a_i and b_i define lateral constraints.

Traditionally, evolutionary algorithms produce different generations of individuals. Each individual is characterized

* Corresponding author. Tel.: +34 94 4137340; fax: +34 93 4016517.

E-mail addresses: Gabriel.bugeda@upc.edu (G. Buggeda), jjrodena@mcm.upv.es (J.J. Ródenas), onate@cimne.upc.edu (E. Oñate).

by a set of values of \mathbf{x} that correspond to a specific structural design. The definition of each design in terms of the values of \mathbf{x} is called the “parametrization” of the optimization problem. The resolution of the optimum design problem consists of finding the values of \mathbf{x} defining the best design. In this work we deal with classical shape optimization in which the values of \mathbf{x} define the geometrical shape of the problem boundary. Alternative optimization approach like topological optimization are not considered here.

In order to take decisions about the quality of each design it is necessary to compute the corresponding values of the objective function and the restrictions. This is carried out by performing a complete structural analysis for each individual, using a specific finite element mesh.

It is well accepted that evolutionary methods are a very powerful and robust tool for the solution of general optimization problems with the ability of not getting trapped in local minima, like in the case of deterministic methods. This makes evolutionary methods very suitable to be used in the context of structural shape optimization methods where, typically, the constraints applied over the maximum values of displacements and stresses produce a lot of local minima where traditional deterministic methods can be trapped. Nevertheless, the use of evolutionary methods requires, typically, the analysis of a very significant number of different designs this requiring a very high computational cost and preventing a more extended use of these methods.

In the structural analysis of each different design the computational cost and the quality of the solution are very much dependent on the quality of the used finite element mesh. One important ingredient of the numerical analysis is the strategy for the generation of a proper mesh for each design. Here we can see two types of strategies:

- To adapt a single existing mesh to the geometries of all different designs. This produces bad shaped and overlapped elements when there are big differences between the geometries of the different designs. Some existing strategies (see Ref. [1]) allow adapting an existing mesh for very big modifications of the boundary shape preventing the elements from being too much distorted. Nevertheless, despite the fact that this type of strategies provides a valid mesh for each design, there is no control of the discretization error contained in the results of each analysis.
- To perform a classical adaptive remeshing procedure for the analysis of each different design. Of course, this procedure ensures good quality results in the numerical analysis of each design, but the total computational cost grows significantly because each design is computed more than once.

This work presents a new strategy that allows generating an adapted mesh for each design without the necessity of performing a full adaptive remeshing procedure. It is based

on the use of sensitivity analysis of all magnitudes related with adaptive remeshing (location of nodes, error estimation, etc.) with respect to the design variables. This sensitivity analysis is performed only once, using a geometry of reference, and it is used to project the results of the corresponding analysis to all other designs to be analyzed. The projected information allows to generate an appropriate adapted mesh for each new design in one shot, greatly reducing the computational cost compared with traditional adaptive remeshing operation over each design. This method is inspired by a similar strategy that was developed and used in the context of deterministic optimization methods (see [2,3]).

In the proposed method, a specific design is selected as a *reference* of the whole population for each of the generations obtained during the evolutionary optimization process. A complete exact sensitivity analysis is performed for this *reference*, including the sensitivities of an error estimator and all the magnitudes involved in a standard adaptive remeshing process. These sensitivities are used to project all the information related with the adaptive remeshing process to all the individuals of a population, thereby allowing the generation of an adapted mesh for each of them in one shot.

It could be argued that if the sensitivity analysis of the numerical solution of each design is performed, it could be natural to use a deterministic optimization strategy instead of an evolutionary one. Nevertheless, it is well known that when there are local minima the deterministic methods can get trapped on them and, in this cases, the evolutionary ones are able to escape from that. This work shows that even when evolutionary methods are used the sensitivity analysis can provide good help in the analysis of each design.

On the other side, it is also well known that deterministic methods can be combined with evolutionary ones by looking for the exact minimum once the evolutionary method has provided a solution close to it. In these cases, the sensitivity analysis is already available and the use of the strategy proposed in this work is straightforward.

2. The proposed methodology

A scheme of the proposed strategy is presented in Fig. 1. For each population, this strategy is summarized as follows:

- A specific individual is selected as a *reference* of all of the population. This individual can be fixed at the very beginning of the optimization process as an initial design. Nevertheless, the best results have been obtained by defining a different *reference* for each generation of the optimization process using the mean values of the design variables of all of the individuals of that particular generation. Thus, if we have a population with P individuals, the values of \mathbf{x}^r that define the *reference* are computed as

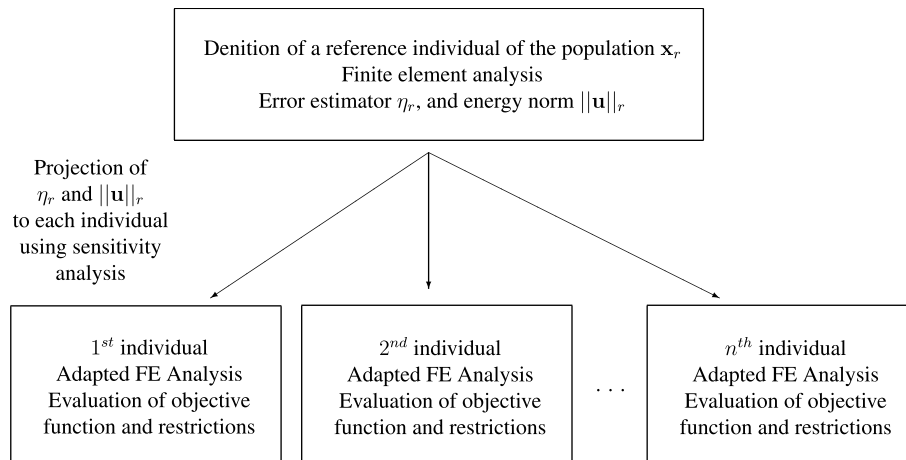


Fig. 1. General scheme of the proposed methodology.

$$\mathbf{x}^r = \frac{1}{P} \sum_{p=1}^P \mathbf{x}^p \quad (1)$$

where \mathbf{x}^p indicates the values of the design variables \mathbf{x} corresponding to the p th individual using the selected parametrization.

- The *reference* design is analyzed using a traditional finite element approach, including an adaptive remeshing strategy to ensure a good quality of the results.
- Once a good mesh is obtained for the analysis of the *reference*, a complete sensitivity analysis of all the magnitudes involved in the adaptive remeshing strategy is computed.
- For each of the individuals to be analyzed, the values of all magnitudes involved in an adaptive remeshing strategy are projected from the *reference*, using the corresponding sensitivity analysis. This projection provides, without any additional computation, an approximation to the values that would be obtained for each specific individual if it would have been computed with a finite element mesh like the one used for the *reference*.
- An adapted mesh is generated for each individual using the projected values of the last step. This adapted mesh is used for the evaluation of the objective function and the restrictions of the individual.

Details of each of the mentioned steps are provided in the following sections.

3. Parametrization of the problem

Each design geometry is represented by using *definition points* which specify some interpolation curves. The curves used here are parametric B-splines. The general expression of a closed B-spline for q points is (see, for instance, Ref. [4]):

$$\mathbf{r}(t) = \sum_{l=0}^q \mathbf{r}_l N_{4,l+1}(t) \quad (2)$$

where $\mathbf{r}(t)$ is the position vector depending on a parametric variable t , and $N_{4,l+1}$ are the normalized fourth order (cubic) B-splines. The curve is expressed as a linear combination of $q+1$ normalized B-splines [4]. The \mathbf{r}_l coefficients are the coordinates of the so-called *polygon definition points* and they are found by using the coordinates of the definition points (see Fig. 2). The degree of continuity of a cubic B-spline is C^2 . By using Eq. (2), the coordinates of the definition points and some additional conditions about slopes and curvatures, the following system of equations can be derived:

$$\mathbf{V} = \mathbf{N}\mathbf{R} \quad (3)$$

where \mathbf{V} is a vector containing the imposed conditions at the definition points, \mathbf{N} is a matrix containing some terms corresponding to the values of the polynomials that define each B-spline, and the \mathbf{R} vector contains the coefficients \mathbf{r}_l to be computed. Details of this process can be found in [4].

The first order sensitivities of \mathbf{R} with respect to each design variable x_i are given by:

$$\frac{\partial \mathbf{R}}{\partial x_i} = \mathbf{N}^{-1} \left(\frac{\partial \mathbf{V}}{\partial x_i} - \frac{\partial \mathbf{N}}{\partial x_i} \mathbf{R} \right) \quad (4)$$

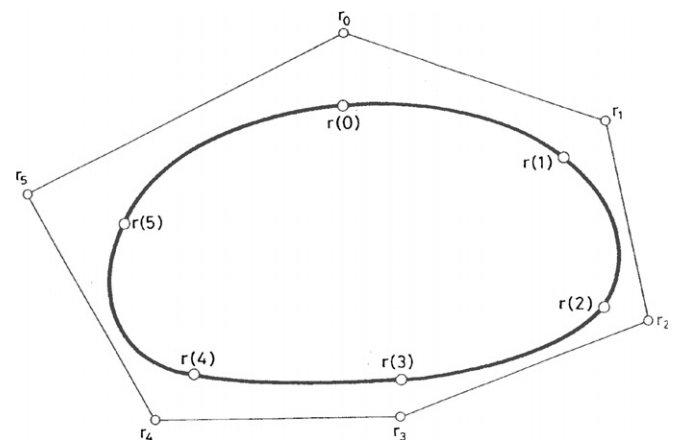


Fig. 2. Parametrization of a curved B-spline.

The derivatives of \mathbf{V} with respect to the coordinates of the definition points chosen as design variables can be easily computed. The vector $\partial \mathbf{R} / \partial x_i$ will contain the term $\partial \mathbf{r}_i / \partial x_i$.

Finally, the sensitivities of the coordinates of any point on the interpolation curve corresponding to a fixed value of t are obtained by:

$$\frac{\partial \mathbf{r}(t)}{\partial x_i} = \sum_{l=0}^q \frac{\partial \mathbf{r}_l}{\partial x_i} N_{4,l+1}(t) \quad (5)$$

4. Mesh generation and mesh sensitivity analysis

The mesh generation algorithm used in this work is the well known advancing front method. This technique is ideal to generate non-structured triangular meshes (see, for instance, Refs. [5,6]).

The characteristics of the desired mesh are specified via a background mesh over which nodal values of the size parameter δ are defined and interpolated using the shape functions. The background mesh for the first design has to be defined by hand. For subsequent designs the background mesh will coincide with the mesh projected into this design from the previous one. This projection will be described later.

Once the sensitivities of the coordinates of each boundary node are known, it is also possible to compute the sensitivities of the coordinates of each internal nodal point (mesh sensitivities). A description and comparison of methods commonly used to evaluate mesh sensitivities can be found in [7]. These sensitivities are used to assess how the mesh evolves when the design variables change.

There are many different ways to define the evolution of the mesh in terms of the design variables. It is possible to consider a simple analogous elastic medium, defining the mesh on movement. This is the case of the “spring analogy”, where each element side is regarded as a spring connecting two nodes. The force produced by each spring is proportional to its length. The solution of the equilibrium problem in the spring analogy is simple but expensive and it involves to solve a linear system of equations with, for 2D problems, two degrees of freedom per node.

In this work, the spring analogy problem has been solved iteratively using a Laplacian smoothing. This technique is frequently used to improve the quality of non-structured meshes. It consists in the iterative modification of the nodal coordinates of each interior node by placing it at the center of gravity of adjacent nodes. The expression of the new position vector of each node \mathbf{r}_i for each iteration is given by:

$$\mathbf{r}_i = \frac{\sum_{j=1}^{k_i} \mathbf{r}_j}{k_i} \quad (6)$$

where \mathbf{r}_j are the position vectors of the k_i nodes connected with the i th node.

The solution of the spring analogy problem with a prescribed error tolerance requires to check the solution after

each smoothing cycle. Taking into account that the described iterative process is only a way to obtain mesh sensitivities, rather than the solution of the equilibrium problem itself, rigorous convergence conditions are not needed. For this reason the number of smoothing cycles to be applied can be fixed a priori. In the examples presented below we have checked that 50 iterations are enough to ensure a good quality of results.

The first-order and higher-order mesh sensitivity analyses along any design variable, x_i , are obtained by differentiating Eq. (6) with respect to x_i for each cycle, i.e.

$$\frac{\partial \mathbf{r}_i}{\partial x_i} = \frac{\sum_{j=1}^{k_i} \frac{\partial \mathbf{r}_j}{\partial x_i}}{k_i} \quad (7)$$

5. Finite element analysis

In this paper we will consider only the solution of structural problems governed by the standard elliptic equations (see, for instance, [8]):

$$\mathbf{L} \mathbf{u} \equiv \mathbf{S}^T \mathbf{D} \mathbf{S} \mathbf{u} = \mathbf{v} \text{ in } \Omega \quad (8)$$

with appropriate boundary conditions. Discretization of Eq. (8) leads to the standard linear system of equations:

$$\begin{cases} \mathbf{K} = \sum_e \mathbf{K}_e \\ \mathbf{K}_e = \int_{\Omega_e} \mathbf{B}^T \mathbf{D} \mathbf{B} d\Omega \\ \mathbf{q} = \sum_e \mathbf{q}_e \\ \mathbf{q}_e = \mathbf{q}_{\Omega_e} + \mathbf{q}_{\Sigma_e} + \mathbf{q}_{P_e} \\ \mathbf{q}_{\Omega_e} = \int_{\Omega_e} \mathbf{N}^T \mathbf{b} d\Omega \\ \mathbf{q}_{\Sigma_e} = \int_{\Gamma_e} \mathbf{N}^T \mathbf{t} d\Gamma \\ \mathbf{q}_{P_e} = \sum \mathbf{N}^T \mathbf{p} \end{cases} \quad (9)$$

where \mathbf{K} , \mathbf{a} and \mathbf{q} denote, as usual, the stiffness matrix, the nodal displacements vector and the equivalent nodal forces vector. Vectors \mathbf{b} , \mathbf{t} and \mathbf{p} correspond to the body, surface and point loads, respectively. Matrix $\mathbf{B} = \mathbf{S} \mathbf{N}$ is used to obtain the strains at each point as $\boldsymbol{\varepsilon} = \mathbf{B} \mathbf{a}$ and the constitutive matrix \mathbf{D} relates strains with stresses as $\boldsymbol{\sigma} = \mathbf{D} \boldsymbol{\varepsilon}$.

6. Error estimation

The error associated with each finite element solution is evaluated for each element using the Zienkiewicz and Zhu [9] error estimator as:

$$\|\mathbf{e}\|_{E_e}^2 \approx \eta_e^2 = \int_{\Omega_e} (\boldsymbol{\sigma}^* - \boldsymbol{\sigma})^T \mathbf{D}^{-1} (\boldsymbol{\sigma}^* - \boldsymbol{\sigma}) d\Omega \quad (10)$$

In this work, nodal stresses $\bar{\boldsymbol{\sigma}}^*$ are recovered using a global least squares smoothing technique (see [9]):

$$\begin{cases} \sigma^* = \sum N_i \bar{\sigma}_i^* = \bar{\mathbf{N}}^T \bar{\sigma}^* \\ \bar{\sigma}^* = \mathbf{M}^{-1} \Phi \\ \Phi = \sum_e \int_{\Omega_e} \bar{\mathbf{N}}^T \sigma \, d\Omega \\ \mathbf{M} = \sum_e \mathbf{M}_e \\ \mathbf{M}_e = \int_{\Omega_e} \bar{\mathbf{N}} \bar{\mathbf{N}}^T \, d\Omega \end{cases} \quad (11)$$

Other procedures for nodal stress recovery, such as the SPR proposed by Zienkiewicz and Zhu [10], can also be used (see for example [11–13]).

The global error estimator η^2 is found by addition of all the elemental contributions $\eta^2 = \sum_e \eta_e^2$. The energy norm can be defined and estimated from the expression:

$$\begin{aligned} \|\mathbf{u}\|_E^2 &= \int_{\Omega} \sigma^T \mathbf{D}^{-1} \sigma \, d\Omega \approx \int_{\Omega} \sigma_{fe}^T \mathbf{D}^{-1} \sigma_{fe} \, d\Omega + \eta^2 \\ &= \mathbf{a}^T \mathbf{K} \mathbf{a} + \eta^2 \end{aligned} \quad (12)$$

This error estimator has been found to be quite robust, reliable and inexpensive. The element sizes for a new mesh are obtained using an adequate remeshing strategy (see, for instance, [14–16]). This issue will be dealt with in the next section.

7. Sensitivity analysis of the error estimator

The error estimator is related to strains or stresses, which are related to displacements. Therefore, the sensitivity analysis of the error estimator requires the computation of the sensitivities of all the magnitudes involved in the analysis.

The objective function will normally be related to the volume of the structure. Constraints are typically related to strains, stresses or displacements. Therefore, it is necessary to evaluate the sensitivities of the structural volume, strains, stresses and displacements to compute the sensitivities of the objective function and the restrictions. This requires the computation of the sensitivities of all the magnitudes involved in the analysis.

The exact sensitivity analysis of all the element expressions can be obtained by direct derivation of Eq. (9). This provides the sensitivities of all magnitudes in terms of the mesh sensitivities previously obtained (details of this process are described in Ref. [17]). The sensitivities of an integral expression are computed after its transformation into the isoparametric domain, whose shape does not depend on the design variables. The Jacobian $|\mathbf{J}|$ of this transformation can be expressed in terms of the nodal coordinates, so that it can also be differentiated in order to know the integral sensitivities. Using the techniques developed in [17], the sensitivities of the element stiffness matrix can be obtained as:

$$\begin{aligned} \frac{\partial \mathbf{K}_e}{\partial x_i} &= \int_{\Omega_e} \left[\frac{\partial \mathbf{B}^T}{\partial x_i} \mathbf{D} \mathbf{B} |\mathbf{J}| + \mathbf{B}^T \frac{\partial \mathbf{D}}{\partial x_i} \mathbf{B} |\mathbf{J}| + \mathbf{B}^T \mathbf{D} \frac{\partial \mathbf{B}}{\partial x_i} |\mathbf{J}| \right. \\ &\quad \left. + \mathbf{B}^T \mathbf{D} \mathbf{B} \frac{\partial |\mathbf{J}|}{\partial x_i} \right] d\xi_1 d\xi_2 \end{aligned} \quad (13)$$

where the sensitivity of the Jacobian is:

$$\frac{\partial |\mathbf{J}|}{\partial x_i} = |\mathbf{J}| \operatorname{tr} \left(\mathbf{J}^{-1} \frac{\partial \mathbf{J}}{\partial x_i} \right) \quad (14)$$

In Eq. (13), matrix \mathbf{B} depends on the nodal coordinates, so that $\partial \mathbf{B} / \partial x_i$ can be obtained from the mesh sensitivities.

Normally, the sensitivities of \mathbf{D} will be zero unless a design variable affects the mechanical properties of the material.

This technique allows to obtain first and higher-order sensitivities of the stiffness matrix \mathbf{K} , the nodal forces vector \mathbf{q} and of any other integral expression involved in the computations. The detailed expressions for the first and higher-order sensitivity analysis can be found in [17].

Eq. (13) allows to obtain the sensitivities of the displacement vector \mathbf{a} as:

$$\frac{\partial \mathbf{a}}{\partial x_i} = \mathbf{K}^{-1} \left[\frac{\partial \mathbf{q}}{\partial x_i} - \frac{\partial \mathbf{K}}{\partial x_i} \mathbf{a} \right] \quad (15)$$

Eq. (15) shows that the inverse of the stiffness matrix is needed for the sensitivity computations. If a direct solver is used this matrix has already been factorized and each new sensitivity analysis involves only a new backsubstitution process. Moreover it is not necessary to assemble the sensitivities of the stiffness matrix because they always appear multiplying a vector and these products can be computed in an element-by-element manner.

The strain and stress sensitivities can be computed as:

$$\frac{\partial \epsilon}{\partial x_i} = \frac{\partial \mathbf{B}}{\partial x_i} \mathbf{a} + \mathbf{B} \frac{\partial \mathbf{a}}{\partial x_i} \quad (16)$$

$$\frac{\partial \sigma}{\partial x_i} = \frac{\partial \mathbf{D}}{\partial x_i} \epsilon + \mathbf{D} \frac{\partial \epsilon}{\partial x_i} \quad (17)$$

The sensitivities of the smoothed stresses are computed in terms of the sensitivities of the mass matrix \mathbf{M} and the Φ vector of Eq. (11). The techniques discussed above for the integral expressions are also used to compute these sensitivities. Finally, the sensitivities of the smoothed stresses are obtained as:

$$\frac{\partial \bar{\sigma}^*}{\partial x_i} = \mathbf{M}^{-1} \left[\frac{\partial \Phi}{\partial x_i} - \frac{\partial \mathbf{M}}{\partial x_i} \bar{\sigma}^* \right] \quad (18)$$

$$\frac{\partial \sigma^*}{\partial x_i} = \bar{\mathbf{N}}^T \frac{\partial \bar{\sigma}^*}{\partial x_i} \quad (19)$$

The same comments about the factorization of the stiffness matrix apply now to the mass matrix.

Following a similar procedure, the first-order sensitivity of the error estimator is obtained from Eq. (10) as:

$$\begin{aligned} \frac{\partial \eta_e^2}{\partial x_i} &= \int_{\Omega_e} \left[\left(\frac{\partial \sigma^*}{\partial x_i} - \frac{\partial \sigma}{\partial x_i} \right)^T \mathbf{D}^{-1} (\sigma^* - \sigma) |\mathbf{J}| \right. \\ &\quad \left. + (\sigma^* - \sigma)^T \frac{\partial \mathbf{D}^{-1}}{\partial x_i} (\sigma^* - \sigma) |\mathbf{J}| + (\sigma^* - \sigma)^T \mathbf{D}^{-1} \left(\frac{\partial \sigma^*}{\partial x_i} - \frac{\partial \sigma}{\partial x_i} \right) |\mathbf{J}| \right. \\ &\quad \left. + (\sigma^* - \sigma)^T \mathbf{D}^{-1} (\sigma^* - \sigma) |\mathbf{J}| \operatorname{tr} \left(\mathbf{J}^{-1} \frac{\partial \mathbf{J}}{\partial x_i} \right) \right] d\xi_1 d\xi_2 \end{aligned} \quad (20)$$

Fuenmayor et al. [18] extended the Zienkiewicz–Zhu [9] error estimator to shape sensitivity analysis in order to develop a discretization error estimator for shape sensitivity analysis, obtaining the expression shown in (20). This proves that the sensitivity of the error is equivalent to the error in sensitivities.

Ref. [19] provides the expressions for the sensitivity analysis of the SPR method. These expressions should replace (16) and (17) in case SPR is used instead of the global smoothing.

In order to use an adaptive mesh refinement strategy, it is also necessary to compute the element and total strain energy. The values of this strain energy and its first-order sensitivities can be approximated from the finite element solution as:

$$\|\mathbf{u}\|_{E_e}^2 \approx \mathbf{a}^T \mathbf{K}_e \mathbf{a} + \eta_e^2 \quad (21)$$

$$\frac{\partial \|\mathbf{u}\|_{E_e}^2}{\partial x_i} \approx \frac{\partial \mathbf{a}^T}{\partial x_i} \mathbf{K}_e \mathbf{a} + \mathbf{a}^T \frac{\partial \mathbf{K}_e}{\partial x_i} \mathbf{a} + \mathbf{a}^T \mathbf{K}_e \frac{\partial \mathbf{a}}{\partial x_i} + \frac{\partial \eta_e^2}{\partial x_i} \quad (22)$$

8. The applied evolutionary algorithm

The adaptive remeshing strategy proposed in this work can be combined with any evolutionary method. For the examples discussed below, Differential Evolution (DE) has been applied. DE is an evolutionary algorithm that has shown a robust performance, yielding good results, even when applied to very different types of problems. Differential Evolution was developed by Storn and Price in 1995 (see [20]). Its key idea is the differential operator, which serves the same purpose as the crossover parameter in a standard genetic algorithm, namely to exchange information between parents when creating offspring.

Outline of the algorithm is shown in Fig. 3.

Storn and Price propose two different versions of Differential Evolution, we use the classical version DE1:

Let $x^p(t)$ be the p th solution parameter vector of generation t :

$$x^p(t) = (x_1^p(t), x_2^p(t), \dots, x_n^p(t)), \quad (23)$$

Outline of the algorithm:

```

begin
  randomly initialize a population P
  evaluate fitness of each individual p ∈ P
  repeat
    for every p ∈ P
      generate trial solution p'
      evaluate(p')
      if fitness(p') > fitness(p)
        then p ← p'
      else
        continue
      end
    end
  until (termination-condition)
end

```

Fig. 3. Outline of the Differential Evolution algorithm.

where n is the length of the parameter vector (in real-value encoding, n corresponds to the number of variables of the objective function). For each vector $x^p(t)$, a so-called trial vector $x^{p'}(t)$ is created by applying the differential operator:

Let A be a subset of $1, 2, \dots, n$. Then for each $j \in A$ holds

$$x_j^{p'}(t) = x_j^q(t) + F(x_j^r(t) - x_j^s(t)), \quad (24)$$

and for each $j \notin A$ holds

$$x_j^{p'}(t) = x_j^p(t), \quad (25)$$

where x_j^q , x_j^r and x_j^s are the j th components of three randomly chosen parameter vectors (with $q, r, s \neq p$ and $q \neq r \neq s$). F is a coefficient usually taken from the interval $(0, 1)$. It controls the amplification of the differential variation.

The size of the subset A determines how many parameters of each solution vector are changed. Storn and Price propose the following scheme for choosing A :

$$A = \{m \div n, (m+1) \div n, \dots, (m+L-1) \div n\} \quad (26)$$

The starting index m is a randomly chosen integer from the interval $[0, n-1]$. L is an integer drawn from the interval $[0, n-1]$ with the probability $P(L=v) = (CR)^v$. $CR \in [0, 1]$ is the crossover probability and constitutes a control variable for the algorithm. Both m and L are chosen anew for each trial vector $x^{p'}(t)$.

The hereby created individual $x^{p'}(t)$ is compared with its parent $x^p(t)$. If it yields a higher fitness, we replace the parent with the trial vector ($x^p(t+1) = x^{p'}(t)$). If not, the original vector is retained ($x^p(t+1) = x^p(t)$).

In DE the crossover operator and the mutation operator have been combined to a single genetic operator, the differential operator. Mutation of a solution can thus only occur within a range defined by the population. If the population converges towards an optimum, the difference between the parameter vectors decreases. Thereby, the magnitude of the mutation is decreased. DE thus incorporates an adaptive stepsize control.

Note, however, that the stepsize is reduced independently of whether the algorithm is converging towards a local or a global optimum. The algorithm only accepts a new solution if it has a higher fitness than its parent. It has no means of escaping, since worse solutions are never accepted. Thus, the algorithm can get stuck in local optima, if the individuals in a generation converge prematurely.

In order to avoid premature convergence, F should not be chosen too low. The threshold depends on the problem at hand. In general, a larger F increases the probability of escaping a local optimum while for $F > 1$ the convergence speed decreases. The crossover probability CR also influences the convergence. A low value speeds up convergence, again posing the problem of premature convergence. A high crossover probability turns the algorithm into a method resembling random search. In this work, $F = 0.85$ and $CR = 0.7$ were chosen.

For a more detailed discussion of Differential Evolution, see also [21].

9. Projection to each design and definition of the adapted mesh

Once the new design has been defined, the new values of the error estimator, the “energy” and the coordinates of the mesh can be projected from the *reference* to each different design as:

$$(x, y)^p = (x, y)^r + \sum_i^n (x_i^p - x_i^r) \left(\frac{\partial x}{\partial x_i}, \frac{\partial y}{\partial x_i} \right) \quad (27)$$

$$\eta^{p^2} = \eta^{r^2} + \sum_i^n (x_i^p - x_i^r) \frac{\partial \eta^2}{\partial x_i} \quad (28)$$

$$\|\mathbf{u}\|_E^p = \|\mathbf{u}\|_E^r + \sum_i^n (x_i^p - x_i^r) \frac{\partial \|\mathbf{u}\|_E^2}{\partial x_i} \quad (29)$$

where x_i^r are the components of the reference vector \mathbf{x}^r defined in expression (1).

The projected values provide the necessary information to perform a remeshing over the next design, even before any new computation is performed. In that sense, the error estimator computed “a posteriori” is transformed into an “a priori” error estimator.

This projection is very important because it allows the quality control of the mesh for each design prior to any new computation. The projected values are used to create the background mesh information needed to generate the mesh for each design.

The generation of every new mesh in the remeshing procedure requires the definition of a “mesh optimality criterion”.

In this work a mesh is considered as optimal when the error density is equally distributed across the volume, i.e. when $\|\mathbf{e}\|_e^2 / \Omega_e = \|\mathbf{e}\|^2 / \Omega$ is satisfied. The justification of this mesh optimality criterion can be found in [14,15].

The combination of the mesh optimality criterion and the error estimation allows to define the new element sizes. Previously, it is necessary to define the admissible global error percentage γ as:

Table 2
Pipe cross section

Design variable	Optimal value
V_1	10.666666
$\sqrt{V_2^2 + V_3^2}$	10.666666
V_4	10.666666

Optimal analytical values of design variables.

Table 3
Pipe cross section

Generation	Area	Error%
Orig. Indiv	413.146	492.53
0	405.698	481.85
1–2	403.353	478.48
3	394.837	466.27
4	391.572	461.59
5	380.297	445.42
6	377.569	441.50
7	368.672	428.74
8	360.197	416.59
9	331.024	374.75
10	328.529	359.70
11	312.700	348.47
12	293.594	321.07
13	287.775	312.72
14	248.683	256.66
15	238.330	241.81
16–17	215.287	208.76
18	183.325	162.92
19	172.175	146.93
20–21	121.327	74.01
22–23	114.472	64.17
24–25	107.505	54.18
26–27	102.514	47.02
28–30	87.954	26.14
31–35	83.300	19.47
36–42	80.041	14.79
43–55	76.814	10.17
56	75.292	7.98
57–65	72.874	4.51
66	72.108	3.42
67–81	71.627	2.73
82	71.140	2.03
83–91	70.676	1.36
92–95	70.469	1.07
96–98	70.334	0.87
99–103	70.175	0.64
104–124	70.152	0.61
125–127	70.146	0.60
128–185	70.049	0.46

Evolution of objective function (area) and difference (error%) with respect to the analytical solution for each generation.

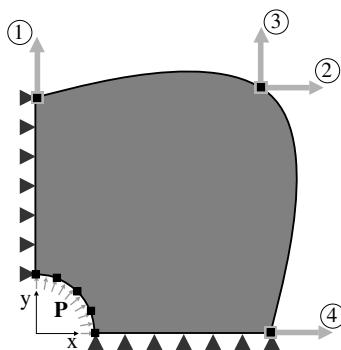


Fig. 4. Pipe cross section. Original model and design variables.

Table 1
Pipe cross section

Design variable	Initial value	Data range	Constraints
V_1	20	[5.2–50.0]	
V_2	19	[4.0–50.0]	
V_3	19	[4.0–50.0]	$V_3 < V_1 - 0.5$
V_4	20	[5.2–50.0]	$V_4 < V_2 + 0.5$

Design variables data.

$$\gamma = 100 \frac{\|\mathbf{e}\|}{\|\mathbf{u}\|} \approx 100 \frac{\eta}{\sqrt{\eta^2 + (\mathbf{a}^T \mathbf{K} \mathbf{a})^2}} \quad (30)$$

The target error level for each element is:

$$\|\mathbf{e}\|_e' = \frac{\gamma}{100} \|\mathbf{u}\| \sqrt{\frac{\Omega_e}{\Omega}} \quad (31)$$

The new element sizes \bar{h}_e can be computed in terms of the old sizes h_e using the expression:

$$\bar{h}_e = \frac{h_e}{\xi_e^{1/p}} \quad \text{with} \quad \xi_e = \frac{\|\mathbf{e}\|_e}{\|\mathbf{e}\|_e'} \quad (32)$$

where p is the order of the shape function polynomials. For further details see [14,15].

Expression (32) provides the size distribution that will be used for the generation of the computational mesh corresponding to each design. This size distribution is computed using the information obtained from the reference individual.

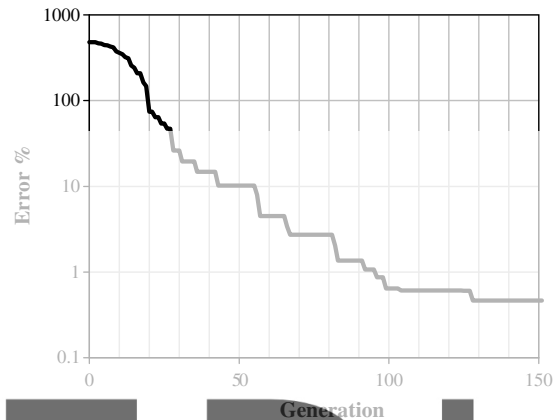
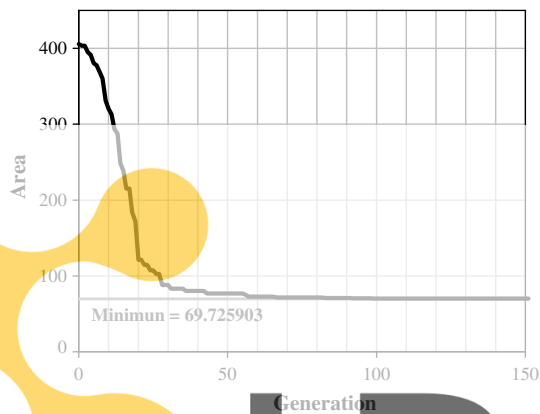


Fig. 5. Pipe cross section. Evolution of area and error with respect to the optimal analytical solution.

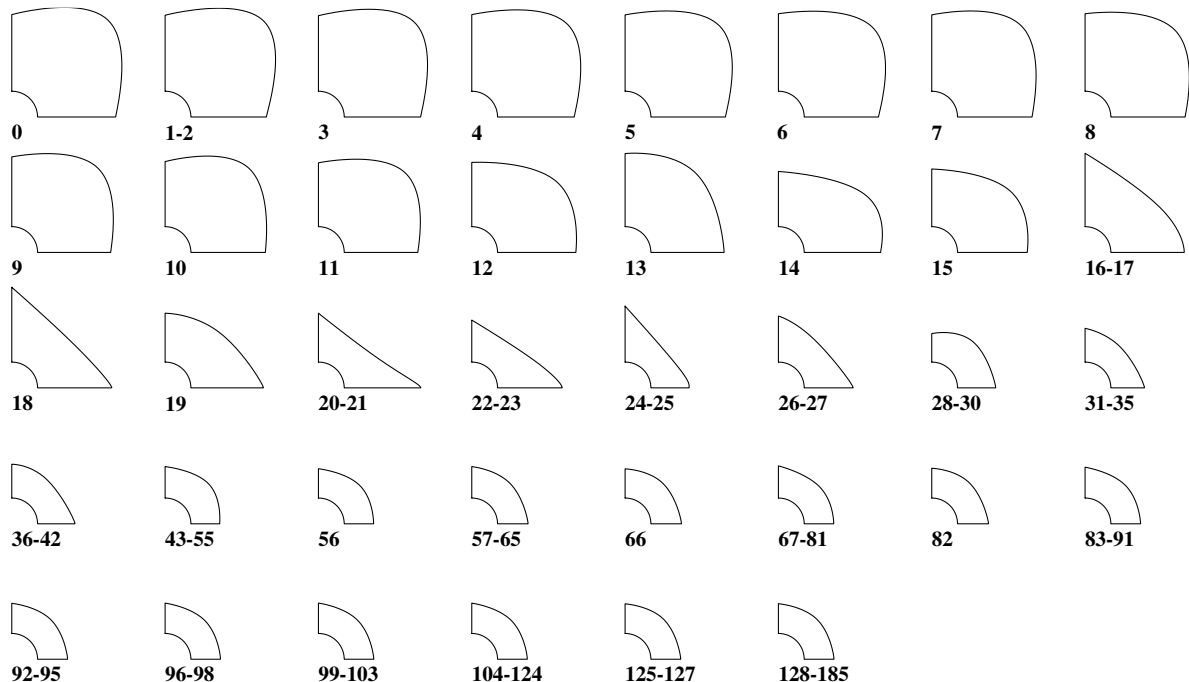


Fig. 6. Pipe cross section. Shape evolution.

Once the analysis of a given design is finished the quality of the obtained results is checked by using the error estimator (10). As expected, in most cases the estimated error will be below the prescribed tolerance, but if it is not the case a new classical adaptive remeshing procedure is used in order to obtain a new set of results with a good quality. The number of cases (designs) in which this operation is needed is very small, as it will be shown in the numerical examples.

10. Numerical examples

Four numerical examples, modelled with six-node quadratic triangular elements, are exposed in this section:

- Pipe cross section.
- Gravity dam.
- Hook.
- Flywheel.

10.1. Pipe cross section

The structure to be optimized corresponds to the cross section of a pipe subjected to an internal pressure. Two perpendicular planes of symmetry were considered, therefore, only one quarter of the section was modeled. The initial shape is shown in Fig. 4. The shape of the component is defined by a total of eight points, five of these points describe a circular internal shape and three describe the exterior of the pipe. The shape optimization problem consists of finding the best shape for the external boundary keeping the internal one fixed. The four design variables used to define the external boundary, were used in the analysis.

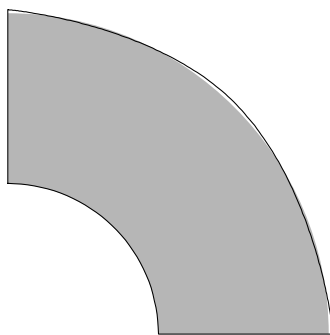


Fig. 7. Pipe cross section. Optimal solution found (black contour) vs. optimal analytical solution (shaded area).

Table 4
Pipe cross section

Design variable	Solution obtained	Analytical solution
V_1	10.764639	10.666666
$\sqrt{V_2^2 + V_3^2}$	10.753775	10.666666
V_4	10.776701	10.666666

Optimal analytical values of design variables vs. optimal solution found.

The mechanical properties considered do not correspond to any specific material. We used $E = 10.1 \times 10^6$ and $\nu = 0.3$. A uniform internal pressure $P = 0.9 \times 10^6$ was applied over the internal surface of radius $R_i = 5$. The objective function is the total cross-sectional area of the pipe. The maximum von-Mises stresses along the boundary of the model were restricted to 2.0×10^6 . The initial values of the design variables and their allowed data range and constraints are shown in Table 1. Observe that we used constraint equations between the values of the design variables in order to minimize the creation of unfeasible individuals.

The area of the analytical solution for this optimization problem (1/4 of cross section) is $A_{\text{opt}} = 69.725903$ that corresponds to an external radius $R_o = 10.666666$. The design variables values that describe the optimal solution are displayed in Table 2.

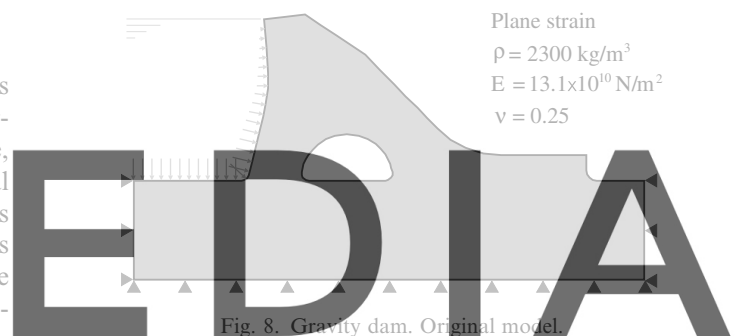


Fig. 8. Gravity dam. Original model.

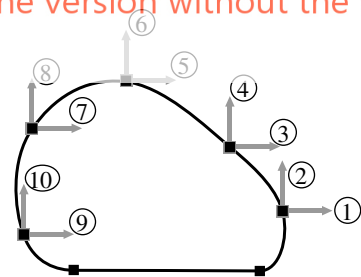


Fig. 9. Gravity dam. Design variables used to define the internal boundary.

Table 5
Gravity dam

Design variable	Initial value	Range	Constraints
V_1	50.0	[50.0–60.0]	
V_2	3.0	[2.0–10.0]	
V_3	47.0	[41.0–50.0]	$V_3 < V_1 - 2.0$
V_4	7.0	[3.0–20.0]	$V_4 > V_2 + 3.0$
V_5	40.0	[25.0–48.0]	$V_5 < V_3 - 3.0$
V_6	10.0	[3.0–25.0]	$V_6 > V_4 + 2.0$
V_7	33.0	[25.0–39.0]	$V_7 < V_5 - 3.0$
V_8	7.0	[3.0–20.0]	$V_8 < V_6 - 2.0$
V_9	30.0	[23.0–30.0]	$V_9 < V_7 - 2.0$
V_{10}	3.0	[2.0–10.0]	$V_{10} < V_8 - 3.0$

Design variables data.

In the analysis, 30 individuals per generation were used. The analysis ran for a total of 185 generations. A maximum estimated relative error in energy norm $\gamma = 1.0\%$ was required for the analysis of each individual.

Table 3 shows the numerical values corresponding to the evolution of the objective function (area) and the difference (error%) between the objective function and the optimal analytical value. These numerical results are presented graphically in Fig. 5.

A graphical representation of the shape evolution as a function of the generation number is shown in Fig. 6.

A comparison between the optimal analytical solution and the solution found at the end of the optimization process is represented in Fig. 7. Table 4 shows the numerical comparison between the optimal values of the design variables and their optimal analytical values taken from Table 2. This comparison clearly shows the high accuracy of the results obtained by means of the proposed methodology.

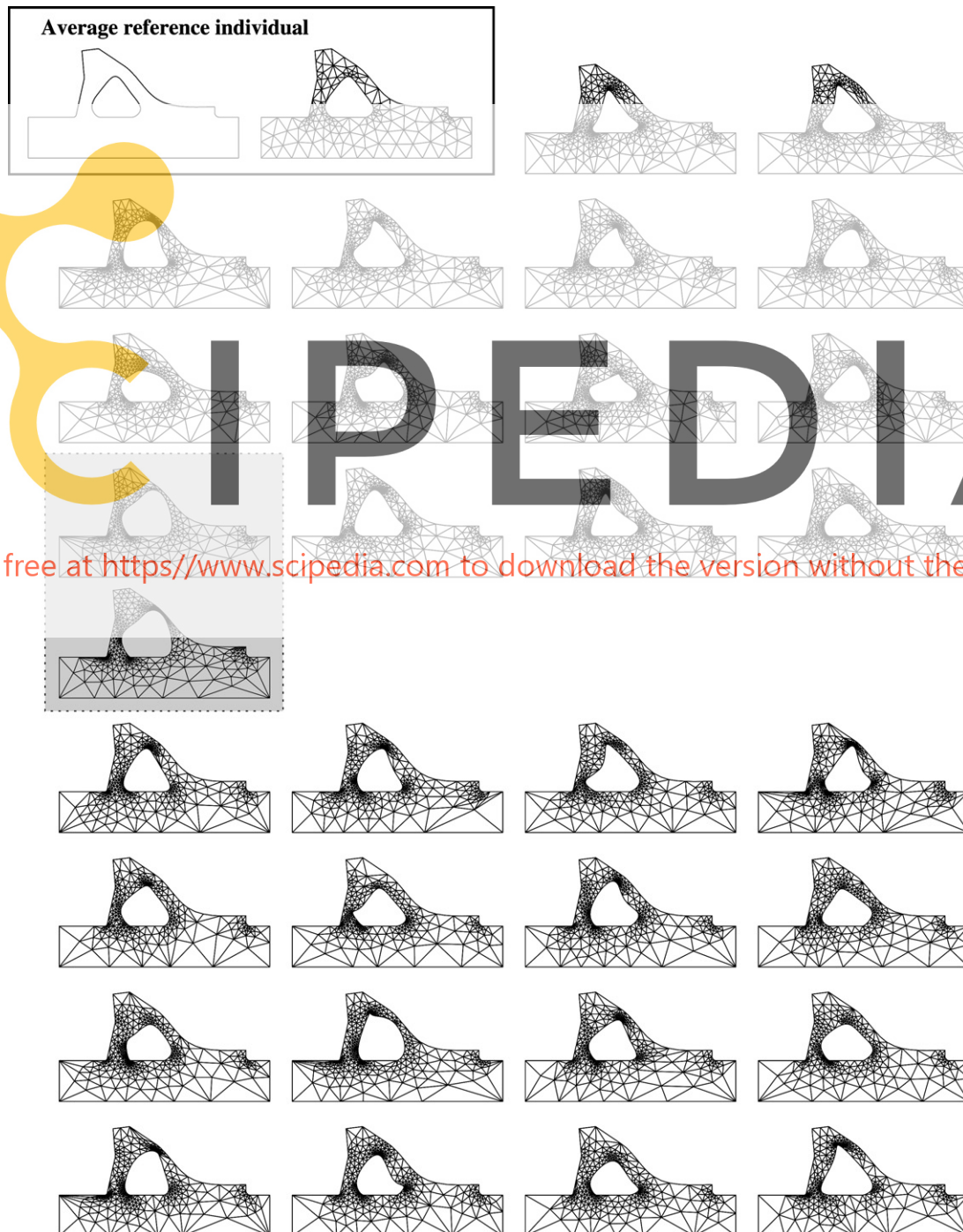


Fig. 10. Gravity dam. Analysis of generation 28. The highlighted individual required remeshing.

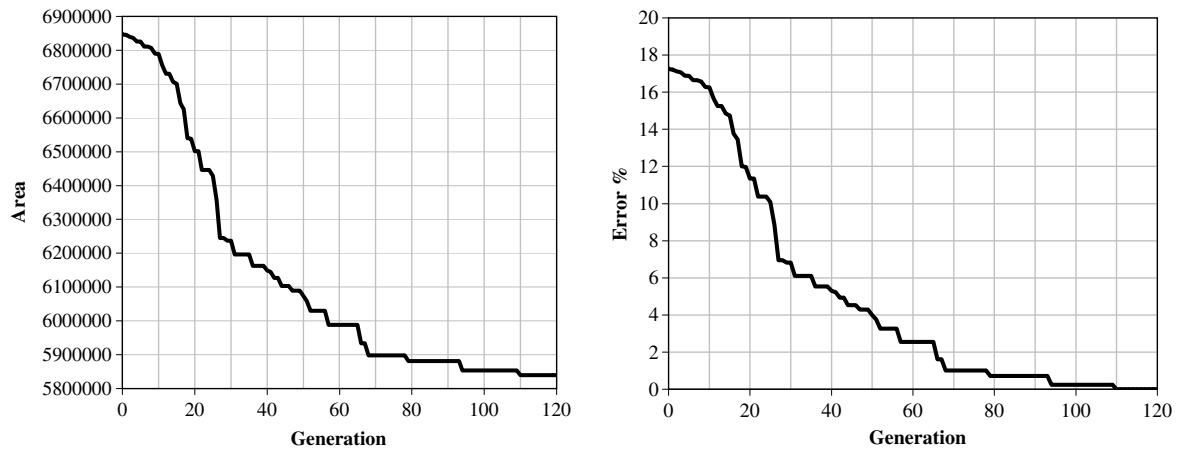


Fig. 11. Dam. Evolution of objective function (area) and difference (error%) with respect to the final solution.

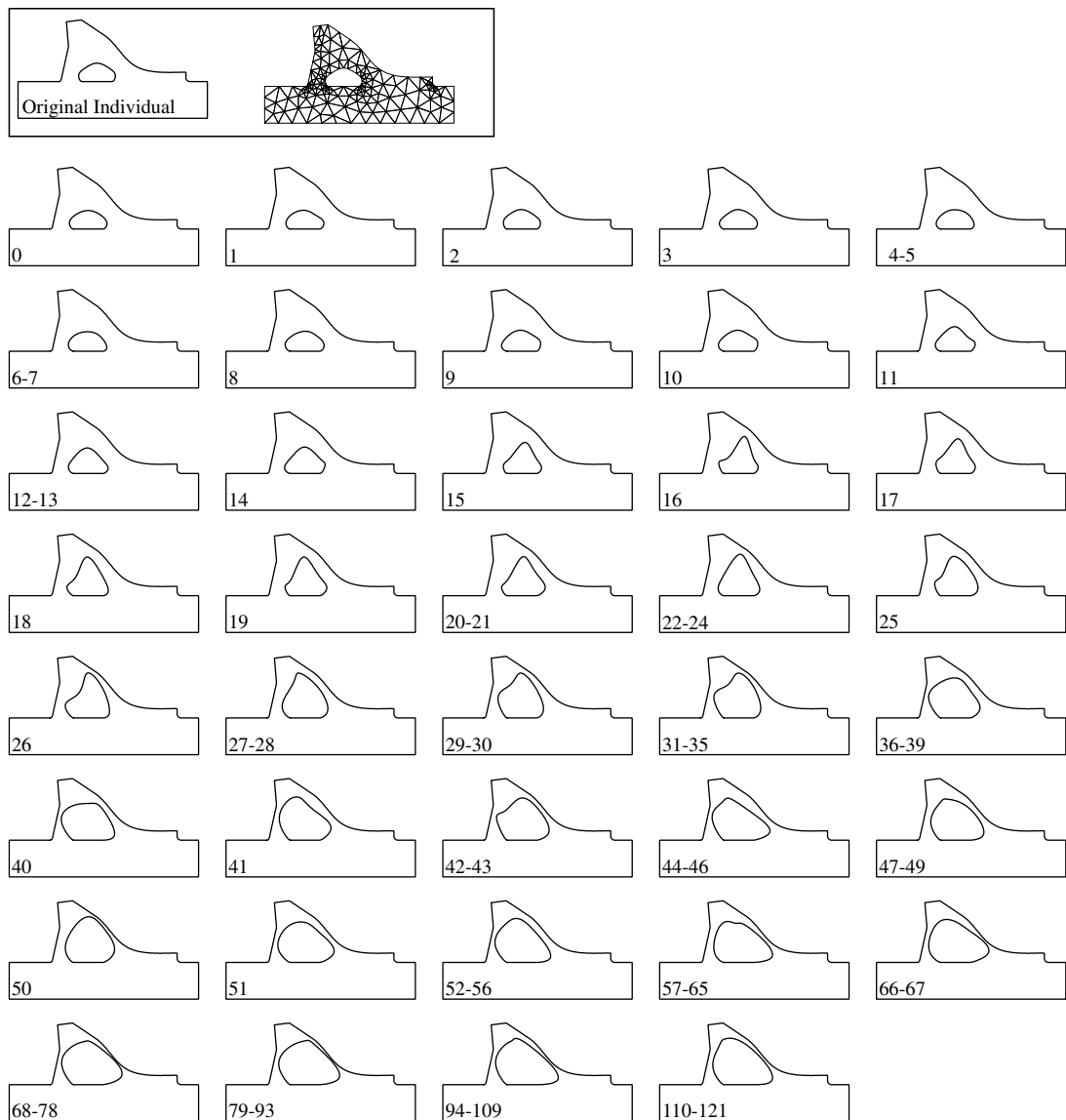


Fig. 12. Dam. Shape evolution.

The difference between the areas corresponding to the analytical solution and the optimal solution found is only 0.46%. In any case, one should take into account that the optimal analytical external contour (circular shape) cannot be obtained because the B-splines used to define the contour are unable to exactly reproduce a circular shape.

For this example, 97% of the meshes obtained with the method of projection from the *reference* proposed in this work provided an estimated relative error in energy norm below the prescribed $\gamma = 1.0\%$. Only the remaining 3% required an additional adaptive remeshing step in order to provide good quality results.

10.2. Gravity dam

The gravity dam represented in Fig. 8 was analyzed. Both dead weight and water pressure were considered in the analysis. The model does not include any sharp reentrant corners.

The shape optimization problem consists of finding the best shape for the internal boundary, keeping fixed the external one, that minimizes the cross-sectional area of the gravity dam. A total of seven points were used to define the internal boundary. As indicated in Fig. 9, the coordinates of five of these points were considered as the design variables for this problem.

The maximum von-Mises stresses along the boundary of the model have been restricted to 2.75×10^6 Pa. The initial values of the design variables and their allowed data range and constraints are shown in Table 5.

In the analysis, 30 individuals per generation were used. The analysis ran for a total of 121 generations. A maximum estimated relative error in energy norm $\gamma = 3.0\%$ was required for the analysis of each individual.

For this example, 94% of the meshes obtained with the method of projection from the *reference* proposed in this work provided an estimated relative error in energy norm below the prescribed $\gamma = 3.0\%$. Only the remaining 6% required an additional adaptive remeshing step for providing good quality results. As an example of the analysis carried out for one generation, Fig. 10 shows the meshes used to analyze the individuals corresponding to the 28th generation. The figure shows the average individual used as *reference* for the analysis of all of the individuals of this generation. In this case, the meshes for 29 of the individuals were directly obtained by means of the data projection from the *reference* individual. Only one of the individuals required remeshing to reduce the estimated error in the energy norm to the specified value.

Fig. 11 shows the evolution of the objective function (area) and the difference (error%) between the objective function and the optimal value obtained at the end of the process. The optimization process provided an area reduction from 6,865,300 to 5,839,540.

A graphical representation of the shape evolution as a function of the generation number is shown in Fig. 12.

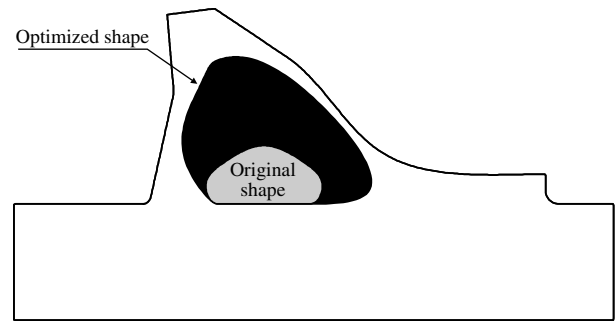


Fig. 13. Dam. Comparison between the initial and the optimized shapes.

Fig. 13 clearly shows the difference between the original and the optimized designs.

10.3. Hook

The proposed methodology is used in this example in the optimization of the shape of a hook in order to minimize its weight. The initial shape, the applied load and the geometry definition points are shown in Fig. 14. The coordinates of 19 points are used as design variables, 16 of them points can move horizontally, one can move vertically and the rest have been enforced to move along straight lines inclined 45° .

A parabolic vertical load has been applied over the inner part of the hook with a resulting load of 630 kg. The material properties are $E = 2,100,000$ kg/cm² and $\nu = 0.3$. A plane stress model has been used. The objective function is the total cross-sectional area of the hook. The maximum von-Mises stresses along the boundary of the model were restricted to 2000 kg/cm². The minimum thickness of the spike of the hook was limited to 0.5 cm.

In the analysis, 20 individuals per generation were used. The analysis ran for a total of 410 generations. A maxi-

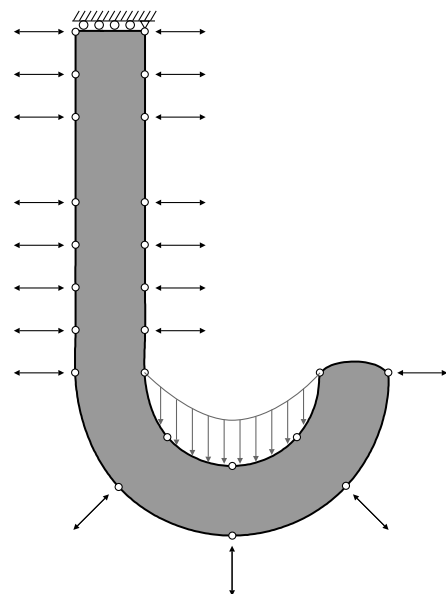


Fig. 14. Hook. Original model and design variables.

mum estimated relative error in energy norm $\gamma = 3.0\%$ was required for the analysis of each individual.

For this example 90% of the meshes obtained with the proposed mesh projection technique provided an estimated relative error in energy norm below the prescribed 3%. The remaining 10% required an additional adaptive remeshing step.

Fig. 15 shows the shape evolution as a function of the generation number. It can be observed that the optimization process displaces the vertical part of the hook until it coincides with the resultant of the load forces in order to eliminate the bending moment over this region. Fig. 16 shows a comparison between the original shape of the hook (black contour) and the final solution obtained (shaded area).

The optimization process provided an area reduction from 181.22 to 101.78 cm². This implies that the original design was 78% heavier than the final solution.

10.4. Flywheel

Consider the design example shown in Fig. 17a in which a fly-wheel is defined with one design region and two non-design regions. A centrifugal load corresponding to a rota-

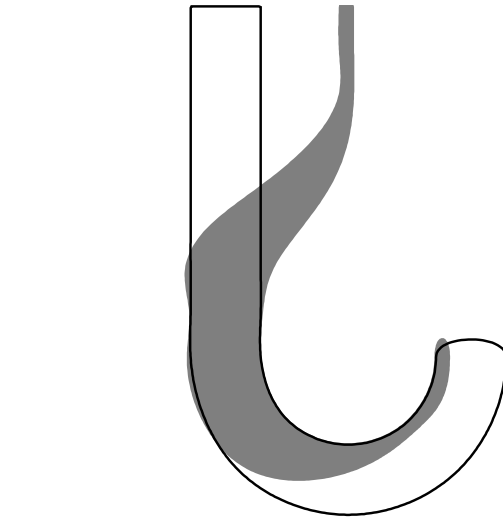


Fig. 16. Hook. Comparison between the initial (black contour) and the optimized (shaded area) shapes.

tion speed of 3 rads/s, and a tangential load corresponding to a distributed load of 1 N/mm² applied over the external boundary are considered. The material properties are: Young's modulus $E = 210$ GPa, Poisson's ratio $\mu = 0.29$ and density $\rho = 7860$ kg/m³.

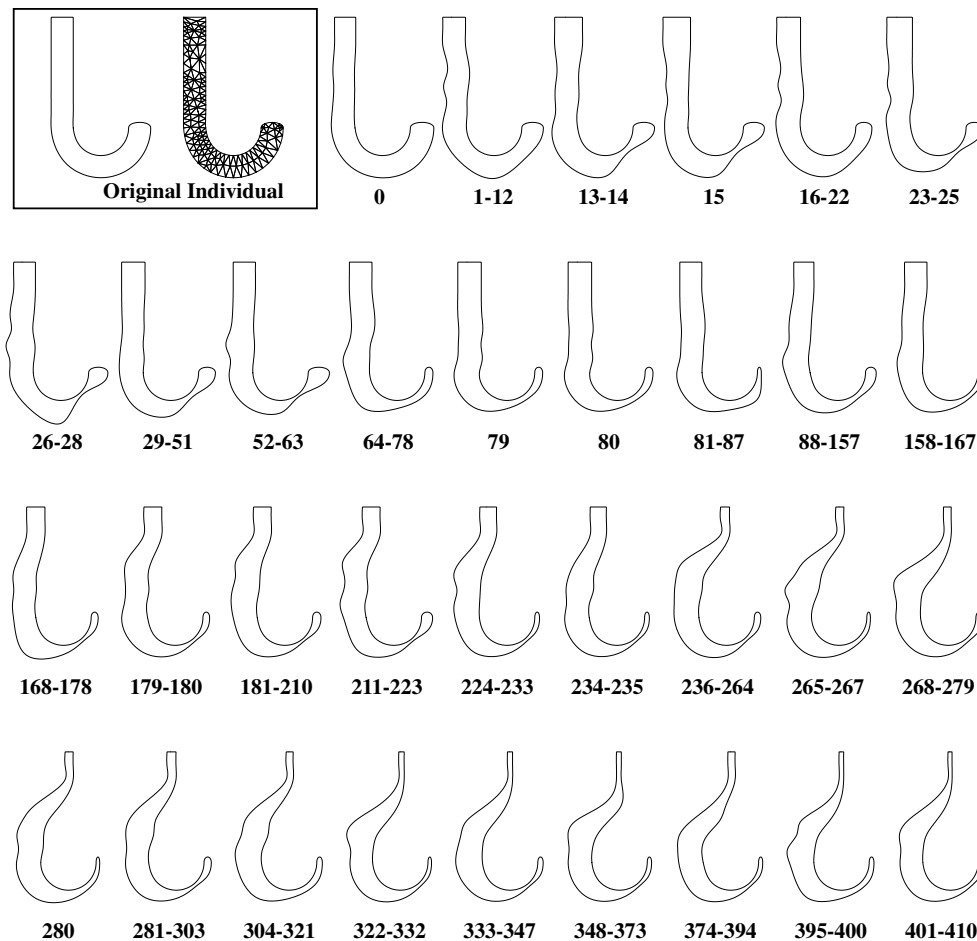


Fig. 15. Hook. Shape evolution.

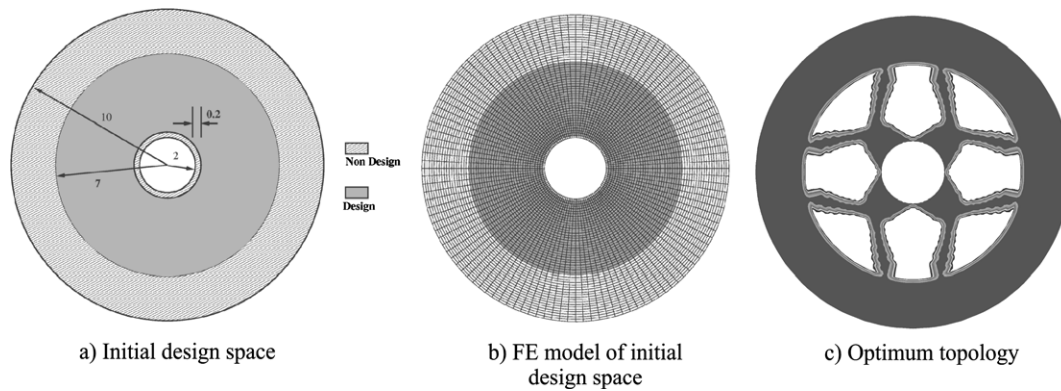


Fig. 17. Flywheel topology optimization.

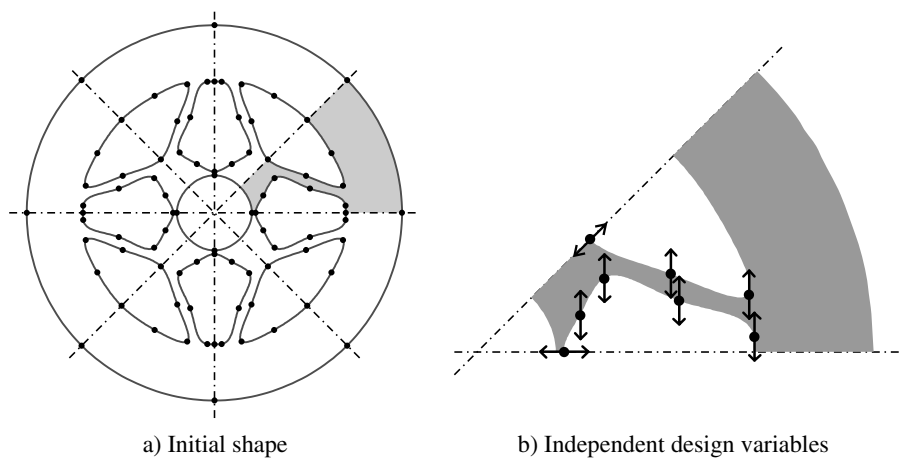


Fig. 18. Flywheel modelization.

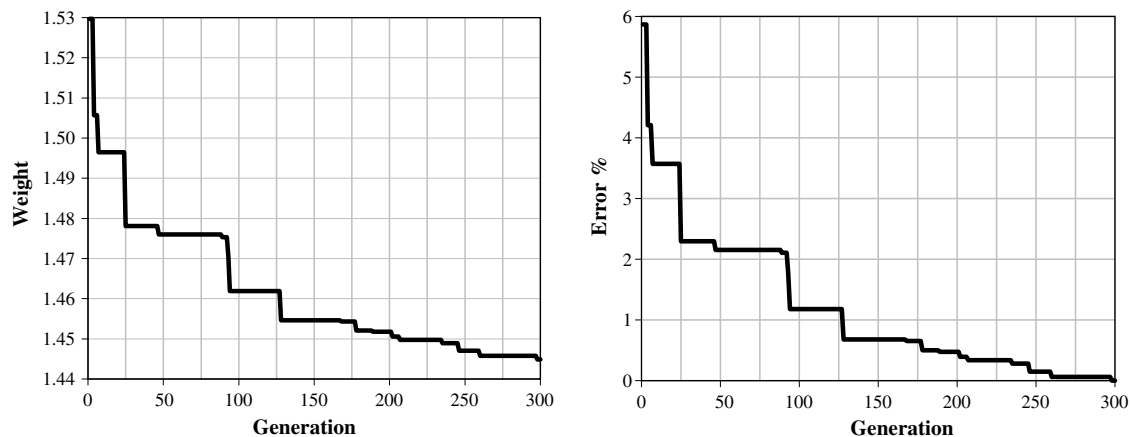


Fig. 19. Flywheel. Evolution of objective function (weight) and difference (error%) with respect to the final solution.

This problem has been previously solved, see [22], using a combination of two optimization tools. In a first step, and from the discretised model represented in Fig. 17b (6000, linear plane stress quadrilaterals), topology optimization was used to find a preliminary layout optimization, obtaining the topology represented in Fig. 17c.

In order to obtain a more refined shape for the flywheel a classical deterministic shape optimization strategy based on shape sensitivity analysis was followed. The initial shape for this analysis, obtained from the preliminary model of Fig. 17c is shown in Fig. 18a. This analysis reduced the total weight from 1.53 to 1.45 kg.

In the analysis presented in this section, the deterministic shape optimization algorithm has been substituted by the evolutionary algorithm previously described.

The geometry of the solution obtained from the topology optimization process has four symmetry axes. However, the tangential load has no symmetry axes and consequently it is not possible to reduce the problem to the study of only one eighth of the total. In any case, in order to limit the computational cost of the shape optimization process, taking into account the presence of the symmetry axes, the following approach has been followed:

- The optimization problem is parameterized using a total of 60 design variables which are some of the coordinates of the key points that are allowed to move.
- Linear constraints were used in order to maintain the symmetries of the model, finally obtaining a total of only eight independent design variables (see Fig. 18b).
- The objective function adopted is the total weight of the flywheel.
- The maximum values of the von-Mises stresses along the boundary were restricted to 100 N/mm^2 .

- 300 generations with 15 individuals per generation were considered.
- A maximum estimated relative error in energy norm $\gamma = 5.0\%$ was required for the analysis of each individual.

Fig. 19 shows the weight values obtained along the evolutionary optimization process and the difference (error%)

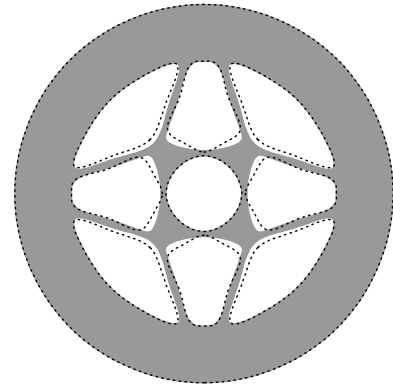


Fig. 21. Flywheel. Comparison between initial and optimized shapes.

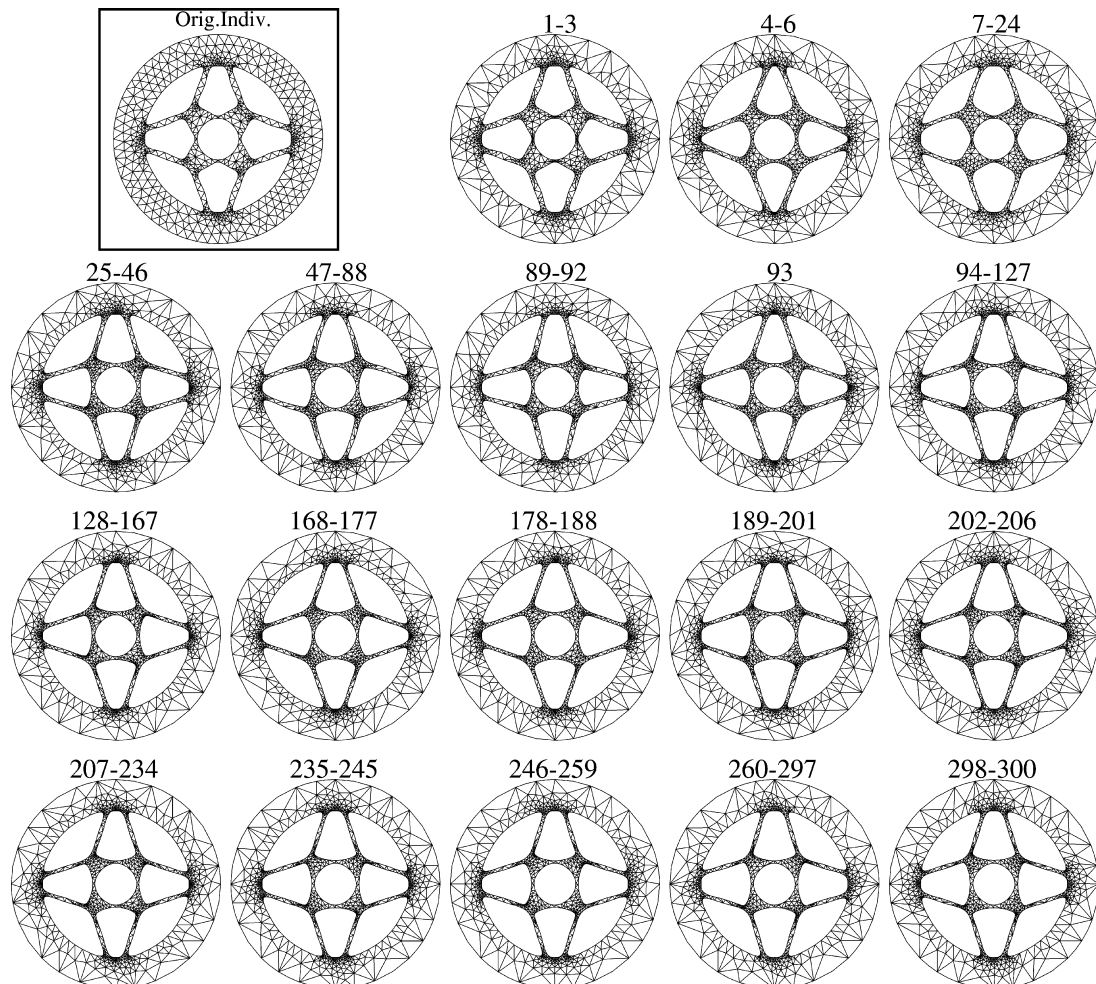


Fig. 20. Flywheel. Shape evolution vs. generation number. Adapted meshes.

between the objective function and the optimal value obtained at the end of the process. A graphical representation of the shape evolution and the corresponding adapted meshes is shown in Fig. 20. The total weight is reduced from 1.53 kg to 1.4445 kg, which is very close to the result obtained in the case of the deterministic optimization process (1.45 kg). Nevertheless, if only the part of the design that is modified is taken into account, then the reduction is from 0.25 kg to 0.17 kg.

Fig. 21 shows a comparison between the original shape (dotted line) and the final solution obtained (shaded area).

11. Conclusions

A strategy for integrating adaptive remeshing procedures into evolutionary optimization processes has been developed and tested. This integration provides an adapted mesh for each design by projecting all necessary information from a *reference* of all the population to each of the different individuals. This projection avoids the necessity of a full adaptive remeshing process over each design.

The proposed strategy provides a control on the quality of the analysis of each design in the cheapest way because only one single analysis is performed for each different individual.

The examples show that the integration of the adaptive remeshing strategy into the evolutionary algorithm does not affect the convergence of the optimization process and ensures a good evaluation of the objective function and the constraints for each different design.

Acknowledgements

The second author has been sponsored by the Generalitat Valenciana and the Universidad Politécnica de Valencia.

The authors want also to thank Mrs. Elke Pahl for her support and assessment in the use of evolutionary algorithms.

References

- [1] Chiandussi G, Bugada G, Onate E. A simple method for automatic update of finite element meshes. *Commun Numer Meth Eng* 2000;16:1–19.
- [2] Bugada G, Oliver J. A general methodology for structural shape optimization problems using automatic adaptive remeshing. *Int J Numer Meth Eng* 1993;18:3161–85.
- [3] Bugada G, Onate E. A methodology for adaptive mesh refinement in optimum shape design problems. *Comput Syst Eng* 1994;5:91–102.
- [4] Faux ID, Pratt MJ. *Computational geometry for design and manufacture*. Chichester: Ellis Horwood; 1987.
- [5] Peraire J. A finite element method for convection dominated flows, Ph.D. thesis, University College of Swansea; 1986.
- [6] Peraire J, Morgan K, Peiró J. Unstructured finite element mesh generation and adaptive procedures for CFD. In: AGARD FDP: specialist's meeting, Loen, Norway; 1989.
- [7] Ródenas JJ, Fuenmayor FJ, Tarancón JE. A numerical methodology to assess the quality of the design velocity field computation methods in shape sensitivity analysis. *Int J Numer Meth Eng* 2004;59:1725–47.
- [8] Zienkiewicz OC, Taylor RL, Zhu JZ. *The finite element method: its basis and fundamentals*. 6th ed. Oxford: Butterworths–Heinemann; 2005.
- [9] Zienkiewicz OC, Zhu JZ. A simple error estimator and adaptive procedure for practical engineering analysis. *Int Numer Meth Eng* 1987;24:337–57.
- [10] Zienkiewicz OC, Zhu JZ. The superconvergent patch recovery and a posteriori error estimates. Part 1. The recovery technique. *Int J Numer Meth Eng* 1992;33:1331–64.
- [11] Wiberg NE, Abdulwahab F, Ziukas S. Enhanced superconvergent patch recovery incorporating equilibrium and boundary conditions. *Int J Numer Meth Eng* 1994;37:3417–40.
- [12] Díez P, Ródenas JJ, Zienkiewicz OC. Equilibrated patch recovery error estimates: simple and accurate upper bounds of the error. *Int J Numer Meth Eng* 2007;69:2075–98.
- [13] Ródenas JJ, Tur M, Fuenmayor FJ, Vercher A. Improvement of the Superconvergent Patch Recovery technique by the use of constraint equations: the SPR-C technique. *Int J Numer Meth Eng* 2007;70:705–27.
- [14] Onate E, Bugada G. A study of mesh optimality criteria in adaptive finite element analysis. *Eng Comput* 1993;104:307–21.
- [15] Bugada G. A comparison between new adaptive remeshing strategies based on point wise error estimation and energy norm error estimation. *Commun Numer Meth Eng* 2002;187:469–82.
- [16] Fuenmayor FJ, Oliver JL. Criteria to achieve nearly optimal meshes in the h-adaptive finite element method. *Int J Numer Meth Eng* 1996;39:4039–61.
- [17] Navarrina F, Bendito E, Casteleiro M. High order sensitivity analysis in shape optimization problems. *Comput Meth Appl Mech Eng* 1989;75:267–81.
- [18] Fuenmayor FJ, Oliver JL, Ródenas JJ. Extension of the Zienkiewicz–Zhu error estimator to shape sensitivity analysis. *Int J Numer Meth Eng* 1997;1413–40.
- [19] Bugada G. A new adaptive remeshing scheme based on the sensitivity analysis of the SPR point wise error estimation. *Comput Meth Appl Mech Eng* 2006;195:462–78.
- [20] Storn R, Price K. Differential evolution – a simple and efficient adaptive scheme for global optimization over continuous space, International Computer Science Institute, Berkeley, CA, USA, Technical Report, March; 1995. <ftp://ftp.icsi.berkeley.edu/pub/techreports/1995/tr-95-012.pdf>.
- [21] Pahl E. Optimization – an attempt at describing the state of the art. CIMNE, Universidad Politecnica de Catalunya, Technical Report, April; 2004.
- [22] Sienz J, Hinton E, Bugada G, Bulman S. Some studies on integrating topology and shape optimization. In: Papadrakakis M, Topping BHV, editors. *Innovative computational methods for structural mechanics*. Saxe-coburg Publications; 1999. p. 223–56.